

CLAIMS

What is claimed is:

1. A method for automatically preserving a read/unread data for a plurality of messages in an email management software when a source database is migrated to a destination database.
2. The method of claim 1 comprising:
 - determining a first universal ID associated with a note ID;
 - translating the note ID into an altered note ID;
 - correlating the location of the altered note IDs to the location of a second universal ID;
 - creating the note ID from the correlation of the location of the altered note ID and the location of the second universal ID; and
 - wherein the first universal ID links the read/unread data to a document data.
3. The method of claim 2 comprising:
 - obtaining a read/unread table from the source database;
 - copying the altered note ID to the destination database; and
 - wherein the read/unread table contains the note ID.
4. The method of claim 3 wherein the first universal ID is located in the source database.
5. The method of claim 4 wherein the second universal ID is located in the destination database.
6. The method of claim 5 wherein the association between the note ID and the document data is the same in the source database as the destination database.
7. The method of claim 6 wherein the above steps occur as part of a synchronization process.
8. The method of claim 7 wherein the note ID points to a fixed location.

9. The method of claim 8 wherein the universal ID points to the document data regardless of whether the document data changes locations within the database.
 10. The method of claim 9 wherein the association between the note ID and the document data is the same in the source database as the destination database even though the arrangement of the document data and the second universal ID have been changed.
 11. The method of claim 10 wherein the note ID is a handle.
 12. The method of claim 11 wherein the size of the note ID is four bytes.
 13. The method of claim 12 wherein the first universal ID is a pointer.
 14. The method of claim 13 wherein the size of the first universal ID is thirty two bytes.
 15. The method of claim 14 wherein the second universal ID is a pointer.
 16. The method of claim 15 wherein the size of the second universal ID is thirty two bytes.
 17. The method of claim 16 wherein the altered note ID is a thirty two byte pointer.
 18. The method of claim 17 wherein the size of the altered note ID is thirty two bytes.
19. A program product for automatically preserving a read/unread data for a plurality of messages in an email management software when a source database is migrated to a destination database, the program product comprising:
- a memory;
 - wherein the memory comprises instructions for a processor to perform steps comprising:
 - determining a first universal ID associated with a note ID;
 - translating the note ID into an altered note ID;

correlating the location of the altered note IDs to the location of a second universal ID;

creating the note ID from the correlation of the location of the altered note ID and the location of the second universal ID; and

wherein the first universal ID links the read/unread data to a document data.

20. The program product of claim 19 when the steps further comprise:

obtaining a read/unread table from the source database;

copying the altered note ID to the destination database; and

wherein the read/unread table contains the note ID.

21. The program product of claim 20 wherein the first universal ID is located in the source database.

22. The program product of claim 20 wherein the second universal ID is located in the destination database.

23. The program product of claim 20 wherein the association between the note ID and the document data is the same in the source database as the destination database.

24. The program product of claim 20 wherein the above steps occur as part of a synchronization process.

25. The program product of claim 20 wherein the note ID points to a fixed location.

26. The program product of claim 20 wherein the universal ID points to the document data regardless of whether the document data changes locations within the database.

27. The program product of claim 20 wherein the association between the note ID and the document data is the same in the source database as the destination database even though the arrangement of the document data and the second universal ID have been changed.

28. The program product of claim 20 wherein the note ID is a handle.
29. The program product of claim 20 wherein the size of the note ID is four bytes.
30. The program product of claim 20 wherein the first universal ID is a pointer.
31. The program product of claim 20 wherein the size of the first universal ID is thirty two bytes.
32. The program product of claim 20 wherein the second universal ID is a pointer.
33. The program product of claim 20 wherein the size of the second universal ID is thirty two bytes.
34. The program product of claim 20 wherein the altered note ID is a thirty two byte pointer.
35. The program product of claim 20 wherein the size of the altered note ID is thirty two bytes.
36. An apparatus for automatically preserving a read/unread data for a plurality of messages in an email management software when a source database is migrated to a destination database, the apparatus comprising:
 - means for obtaining a read/unread table from the source database;
 - means for determining a first universal ID associated with a note ID;
 - means for translating the note ID into an altered note ID;
 - means for copying the altered note ID to the destination database;
 - means for correlating the location of the altered note IDs to the location of a second universal ID;
 - means for creating the note ID from the correlation of the location of the altered note ID and the location of the second universal ID; and
 - wherein the first universal ID links the read/unread data to a document data;
 - wherein the read/unread table contains the note ID;

wherein the first universal ID is located in the source database;

wherein the second universal ID is located in the destination database;

wherein the association between the note ID and the document data is the same in the source database as the destination database;

wherein the above steps occur as part of a synchronization process;

wherein the note ID points to a fixed location;

wherein the universal ID points to the document data regardless of whether the document data changes locations within the database;

wherein the association between the note ID and the document data is the same in the source database as the destination database even though the arrangement of the document data and the second universal ID have been changed;

wherein the note ID is a handle;

wherein the size of the note ID is four bytes;

wherein the first universal ID is a pointer;

wherein the size of the first universal ID is thirty two bytes;

wherein the second universal ID is a pointer;

wherein the size of the second universal ID is thirty two bytes;

wherein the altered note ID is a thirty two byte pointer; and

wherein the size of the altered note ID is thirty two bytes.